AD-A081 476 MARYLAND UNIV COLLEGE PARK COMPUTER VISION LAB F/G 12/1
ALGORITHMS FOR TESTING CONVEXITY OF DIGITAL POLYGONS.(U)
DEC 79 R KLETTE, E V KRISHNAMURTHY AFOSR-77-3271
UNCLASSIFIED TR-841 AFOSR-TR-80-0138 NL

1 OF 1
AD-
A081476

END
DATE
FILMED
4-80
DTIC

1.0 1.1 1.25 1.4 1.6 1.8 2.0 2.2 2.5 2.8 3.2 3.6 4.0 4.5 5.0 5.6 6.3 7.1 8.0 9.0

MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AFOSR-TR- 80-0138

12

LEVEL II

COMPUTER SCIENCE TECHNICAL REPORT SERIES

UNIVERSITY OF MARYLAND
COLLEGE PARK, MARYLAND
20742

80 3 3 020

(18) AFOSR

(19) TR-80-0138

(9) Interim rept.,

(14) TR-841

AFOSR-77-3271

(11) December, 1979

(12) 21

(6) ALGORITHMS FOR TESTING CONVEXITY OF DIGITAL POLYGONS,

(10) Reinhard Klette* ~~and~~ E. V. Krishnamurthy+

Computer Vision Laboratory
Computer Science Center
University of Maryland
College Park, MD 20742

(16) 2304

(17) A2

ABSTRACT

A simple approach based on Shoenberg's theorem is described to test whether a set of border points of a simply 4-connected digital picture is convex. The sequential implementation of this method is linear in the number of points; the parallel algorithm needs constant time only, using bitwise parallel Boolean operations and shifts on binary matrices. Suitable modifications of this approach can be used for decomposing two-dimensional objects into convex sets and for filling concavities.

*Permanent address: Sektion Mathematik, Friedrich-Schiller-Universität Jena, DDR-69 Jena, German Democratic Republic.

+Permanent address: Indian Institute of Science, Bangalore-560012, India.

(15)

The support ~~of the U.S. Air~~ Force Office of Scientific Research under Grant AFOSR-77-3271 is gratefully acknowledged, as is the help of Kathryn Riley in preparing this paper. The first author thanks the government of GDR for financial support. The authors are thankful to Azriel Rosenfeld for continued encouragement and valuable suggestions.

411 074

JB

1. Introduction

In recent years, digital image processing has rapidly grown into a major discipline with applications in a variety of areas [5,6]. One important problem in digital image processing is to test whether a digital object is convex and if not to detect the concavities. There are a number of approaches to this problem [1,2,4,7,9,11-15]. Our aim here is to describe a very simple algorithm to test whether a simply 4-connected digital object S (i.e., an object without holes) has a border which is convex or not. For this purpose we think of S as a set of lattice points in the Euclidean plane and regard it as a polygon, as in [8], with the border points of S as its vertices and having edges of length 1 (between the border points which are horizontally or vertically adjacent) and $\sqrt{2}$ (between the border points which are diagonally adjacent).

The test we use is based upon a very powerful theorem due to Shoenberg [10] which gives necessary and sufficient conditions for a closed polygon to be convex in an even-dimensional Euclidean space. In sequential implementation this approach leads to a border following algorithm which is linear in the number of border points. In parallel implementation the approach leads to a pattern matching algorithm, i.e. a local operation on binary images, which runs within constant time using bitwise parallel Boolean operations and shifts on binary images as basic instructions.

2. Shoenberg's Theorem

Let $\Pi = P_0, P_1, \ldots, P_k$ where $P_i = (x_{i1}, x_{i2}, \ldots, x_{im})$ is a polygon with vertices P_i $(0 \le i \le k)$ in m-dimensional Euclidean space, E_m. We assume that Π spans E_m, which is the case provided the matrix

$$X = ||1, x_{i1}, x_{i2}, \ldots, x_{im}|| \qquad (1)$$

$$(i = 0,1,2,\ldots,k,\ k \ge m)$$

is of rank $(m+1)$.

We say that the polygon Π is <u>convex</u> on E_m provided it spans E_m and crosses no hyperplane more than m times.

Shoenberg [9] proves that the polygon Π is convex on E_m iff the matrix X defined by (1) is of rank $(m+1)$ and all its non-vanishing minors of order $(m+1)$ are of the same sign.

This theorem is intuitively reasonable, as it says that the convexity of Π on E_2 requires that no two triangles $P_\alpha P_\beta P_\gamma$ $(\alpha<\beta<\gamma)$ have opposite orientations.

The parity of m, however, plays a role. If m is even, and Π is convex in E_m, then also the closed polygon (with k+1 sides)

$$\Pi_1 = P_0 P_1 \ldots P_k P_0$$

is convex on E_m. For the case where m is odd it can be shown that the first and last vector of a convex polygon can never coincide. Fortunately, for our application in two dimensions this theorem is very useful.

For the closed polygon Π_1, it does not matter which vertex is taken to be the first as long as the correct cyclic order of the vertices is preserved. This is evident also from the fact that the cyclic permutations of the rows of the matrix X will not change the common sign of its minors of the odd order of (m+1). However, for the sequential implementation of our approach for detecting concavities we start with the border point which is the first from left to right in the uppermost row of the simply connected digital object S.

3. Test for Convexity

The condition of Shoenberg's theorem is very strong (both necessary and sufficient) to test for convexity of polygons. For two-dimensional polygons with (k+1) sides, drawn on a lattice (as in the case of a digital picture), this test takes a very simple form of evaluating k+1 inner products of two three-component vectors.

Let the border points of S forming a closed polygon with (k+1) vertices $(P_0, P_1, \ldots, P_k, P_0)$ be denoted by their position coordinates in the lattice, thus:

$$P_0 = (x_0, y_0);\ P_1 = (x_1, y_1); \ldots;\ P_i = (x_i, y_i); \ldots; P_k = (x_k, y_k).$$

Then according to Shoenberg's theorem, $\Pi = P_0 \ldots P_k P_0$ is a closed convex polygon iff the matrix

$$X = \begin{bmatrix} 1 & x_0 & y_0 \\ 1 & x_1 & y_1 \\ \cdot & \cdot\ \cdot\ \cdot & \cdot \\ 1 & x_i & y_i \\ \cdot & \cdot\ \cdot\ \cdot & \cdot \\ 1 & x_k & y_k \end{bmatrix} \tag{2}$$

is of rank 3 and all its non-vanishing minors are of the same sign.

This test can be carried out as follows:

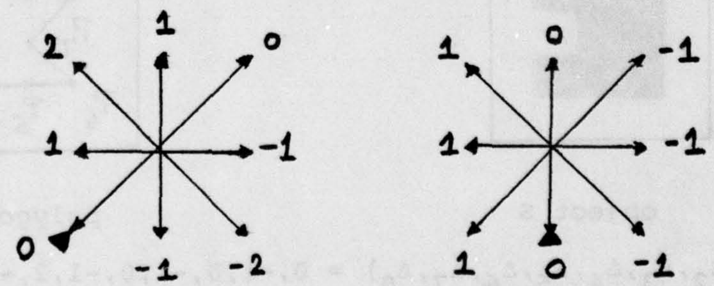
Let

$$\Delta_i = \mathrm{Det} \begin{bmatrix} 1 & x_{i-1} & y_{i-1} \\ 1 & x_i & y_i \\ 1 & x_{i+1} & y_{i+1} \end{bmatrix} \tag{3}$$

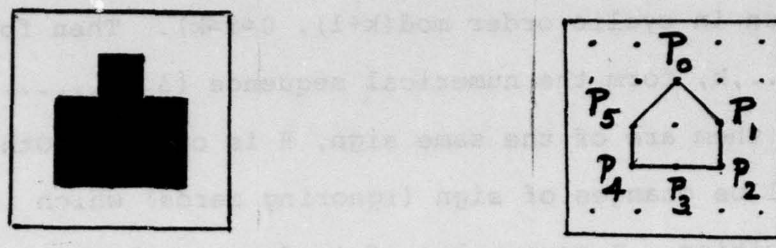
(i is taken in cyclic order mod(k+1), $0 \le i \le k$). Then for each $i = 0,1,\ldots,k$, form the numerical sequence $\{\Delta_1,\Delta_2,\ldots,\Delta_k,\Delta_0\}$. If all of them are of the same sign, Π is convex. Otherwise, there will be changes of sign (ignoring zeros) which indicate the concavities. A zero value of Δ_i denotes that the given three points are collinear (a degenerate triangle).

In Figures 1 and 2 we give an illustration of this procedure. In our figures, ■ denotes an object point, and □ a background point.

However, it is not necessary to perform the inner products of these three-component vectors, i.e., to compute the Δ_i's, during such a test procedure. There exists only a finite number of possible pairs of such vectors. All possible combinations with the corresponding Δ_i-values are included in the following schemes:



(and all 90° rotations of these schemes). In the first step we go from ▶ to the center point of such a scheme; in the second step we go from the center point in a direction which is labeled with value δ. During both steps the interior of the object S (of the polygon Π) is on our right side. Let the center point

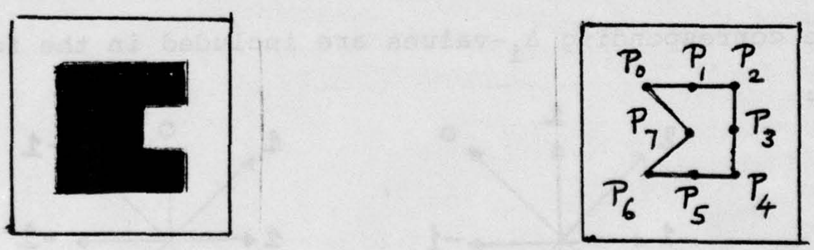


object S polygon Π

$(\Delta_1,\Delta_2,\Delta_3,\Delta_4,\Delta_5,\Delta_0) = (-1,-1,0,-1,-1,-2)$

The object S (polygon Π) is convex.

Figure 1. Example of a convex object.



object S polygon Π

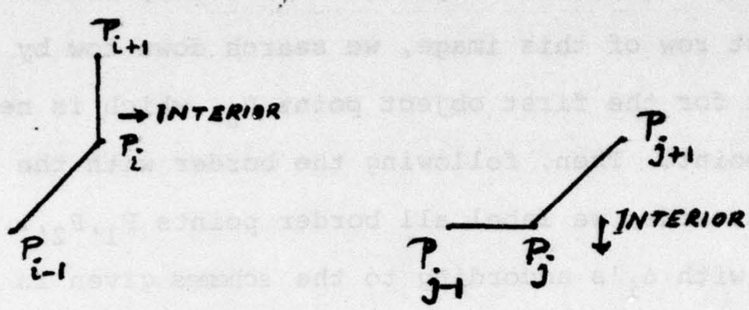
$(\Delta_1,\Delta_2,\Delta_3,\Delta_4,\Delta_5,\Delta_6,\Delta_7,\Delta_0) = 0,-1,0,-1,0,-1,2,-1)$

The object S (polygon Π) is concave; there are two sign changes in the cyclic order indicating one concavity at point P_7.

Figure 2. Example of a concave object.

be the border point P_i. Then $\Delta_i = \delta$. In particular, all possible Δ_i-values are taken from the set $\{-2,-1,0,1,2\}$.

For example in the cases



we have $\Delta_i = 1$ and $\Delta_j = 1$.

4. Sequential Implementation

Let us assume that in one binary image there is exactly one simply 4-connected object S. At first, starting with the uppermost row of this image, we search down row by row from left to right for the first object point P_0, which is necessarily a border point. Then, following the border with the interior on the right side, we label all border points $P_1, P_2, \ldots, P_k, P_0$ step by step with Δ_i's according to the schemes given in Section 3. During this procedure we can count the number of sign changes (if there is any positive Δ_i-value then the object is concave), or the number of border points, or we can add all positive Δ_i-values in one register, say R^+, and all negative Δ_i-values in another register, say R^-. At the end of this border following procedure it is possible to use such results to compute measures of convexity--for example

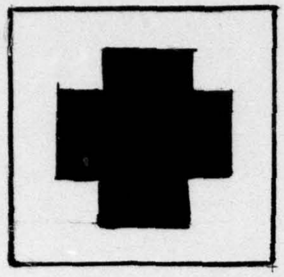
$$C_1(S) = \frac{\text{number of sign changes x 100}}{\text{number of border points}}$$

or

$$C_2(S) = \frac{(R^-+1)\text{x}100}{(R^++1)\text{x number of border points}}$$

In Figure 3 we give four examples of objects and their measures C_1 and C_2. Of course, an object S is convex iff $C_1(S) = 0$.

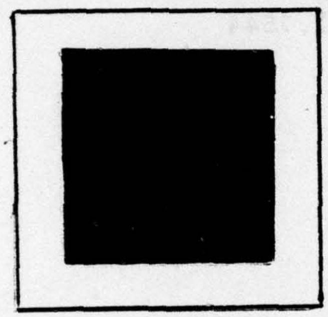
Let us assume the object S is encoded in Freeman code representation (see, e.g., [4]) where the directions are encoded according to the following scheme:



S_1

$C_1(S_1) = 0$

$C_2(S_1) = 112.5000$

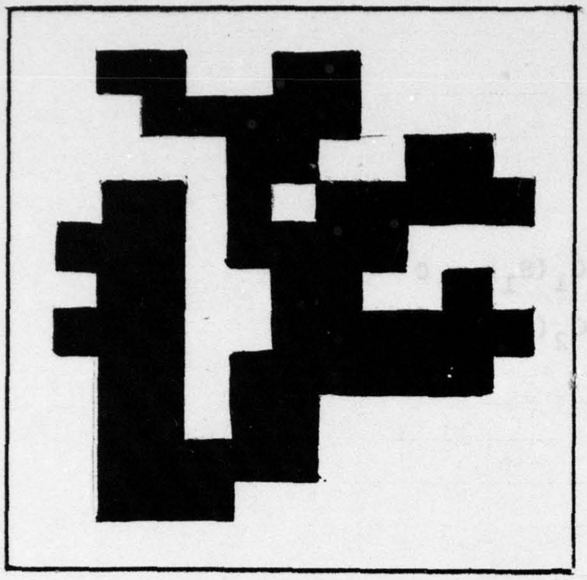


S_2

$C_1(S_2) = 0$

$C_2(S_2) = 31.2500$

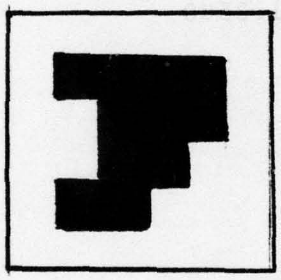
Figure 3. Examples for C_1 and C_2 values.



S_3

$C_1(S_3) = 33.3333$

$C_2(S_3) = 1.7544$

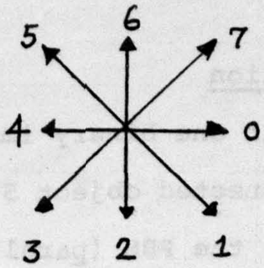


S_4

$C_1(S_4) = 20$

$C_2(S_4) = 20$

Figure 3, continued.



For example, the Freeman code representation for object S_4 in Figure 3 is 0002334765. It is possible to calculate the Δ_i-sequence for a given object S (and related values, like R^+ or R^-) while reading the Freeman code sequence of the object from left to right, using the labeling schemes given in Section 3. For example, for the object S_4 we get

$00 \rightarrow \Delta_1=0$	$23 \rightarrow \Delta_4=-1$	$47 \rightarrow \Delta_7=-1$	$50 \rightarrow \Delta_0=-1$
$00 \rightarrow \Delta_2=0$	$33 \rightarrow \Delta_5=0$	$76 \rightarrow \Delta_8=1$	
$02 \rightarrow \Delta_3=-1$	$34 \rightarrow \Delta_6=-1$	$65 \rightarrow \Delta_9=1$	

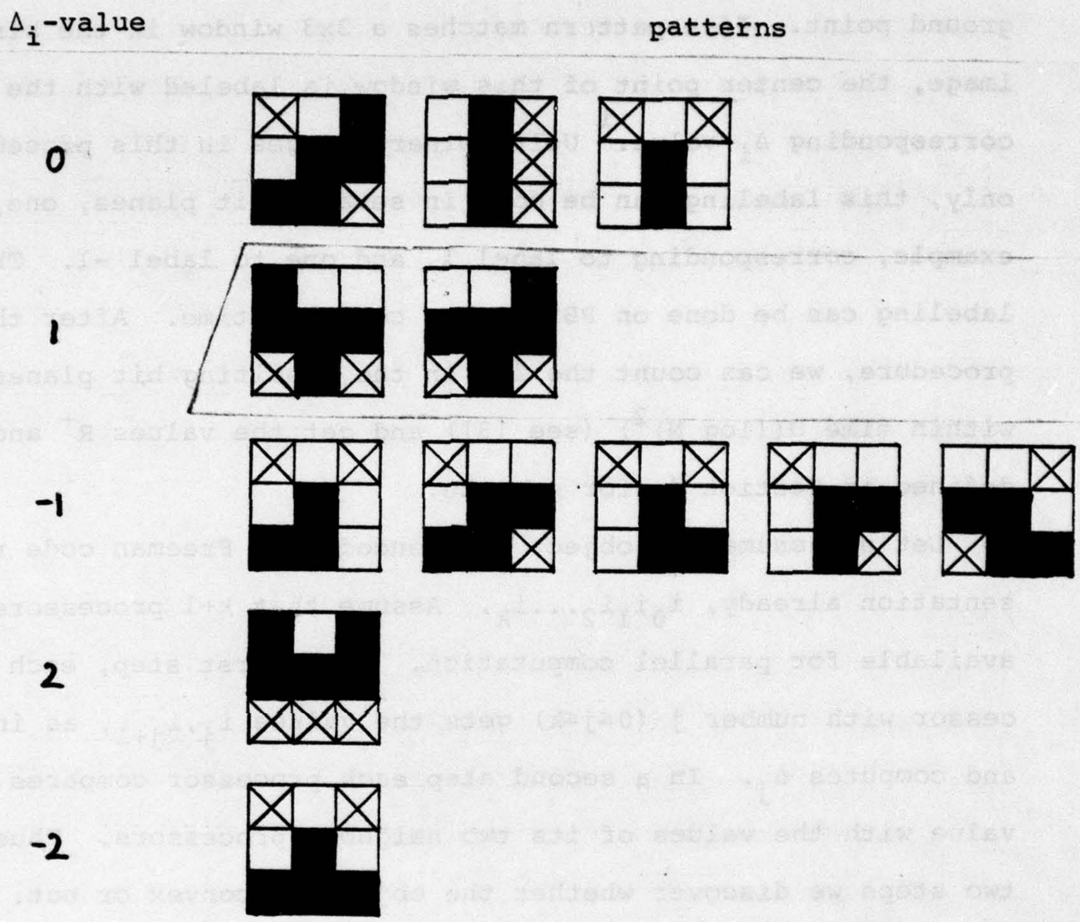
5. Parallel Implementation

Let us assume that in one binary image of size NxN there is exactly one simply 4-connected object S. We choose as a model for parallel computation the PBS (parallel binary image processing system); the exact definition of this model can be found in [3]. The PBS can perform bitwise parallel Boolean operations and shift operations on binary images as basic instructions. For example, the program for computing the edge image for the given object S has the following form:

```
input binary image X of size NxN with object S;
begin binary images Y,Z of size NxN;
      Y=X→1; Z=X∧Y; Y=X←1; Z=Z∧Y;
      Y=X↑1; Z=Z∧Y; Y=Y↓1; Z=Z∧Y;
      Z=Z̄; Y=X∧Z; print Y
end.
```

In [3] it was explained that the PBS is very time efficient for computing local operations on binary images; edge detection is one example of such a local operation. Fortunately, the labeling of the border points of S with Δ_i-values is such a local operation.

Essentially, the computation of the Δ_i-values can be described as a special pattern matching procedure. The necessary size of the patterns is 3x3 only. In Figure 4 we have listed all patterns which must be matched during this procedure. In this figure, ■ represents an object point, □ represents a background



and all 90° rotations of these patterns

Figure 4. Δ_i-patterns.

point, and ☒ can be either, i.e., an object point or a background point. If a pattern matches a 3x3 window in the binary image, the center point of this window is labeled with the corresponding Δ_i-value.[1] Using binary images in this procedure only, this labeling can be done in several bit planes, one, for example, corresponding to label 1, and one to label -1. This labeling can be done on PBS within constant time. After this procedure, we can count the 1's in the resulting bit planes within time $O((\log N)^2)$ (see [3]) and get the values R^+ and R^- defined in Section 4, for example.

Let us assume the object S is encoded in Freeman code representation already, $i_0 i_1 i_2 \ldots i_k$. Assume that k+1 processors are available for parallel computation. In a first step, each processor with number j $(0 \le j \le k)$ gets the values i_j, i_{j+1}, as input and computes Δ_j. In a second step each processor compares its value with the values of its two neighbor processors. Thus, after two steps we discover whether the object is convex or not. The values R^+ and R^-, or the number of sign changes, can be computed using k+1 processors within $\lfloor \log_2 k \rfloor$ steps.

---

[1] In using these masks, we assume that the object thickness is at least two; otherwise, pattern matching with these masks may result in non-unique labels.

6. Concluding Remarks

a. The approach described for detection of concavities can be applied to decompose polygons into convex sets.

b. It can also be used to fill concavities by addition or deletion of edges at the appropriate vertices.

c. An iterative application (relaxation algorithm) of comparisons between Δ_i-values in the neighborhoods of border points can be used for smoothing of "small concavities."

d. After this report was prepared, the authors came across a recent paper by Bribiesca and Guzman [16] in which a method based on Freeman Code is given for the description as well as measurement of differences in shapes (shape taxonomy) among polygonal boundaries--a shape being defined as that property of an object which is invariant under translation, rotation and similarity transformation (choice of grid or lattice size and its orientation). In order to arrive at a canonical description, the orientation of the grid is normalized by taking one of its axes parallel to the length of a basic rectangle which just contains the boundary; however, the choice of grid size still plays a role in uniqueness. For this purpose the following procedure is used: each lattice point along the boundary is encoded in ternary base, viz. 0 for Convex, 1 for Straight, and 2 for Concave. This resulting ternary number is then circular shifted until it assumes the least value; this serves as the shape number; the number of digits chosen naturally depends on the grid size and hence for uniqueness the number of digits (called order) is prescribed.

It is clear that the algorithm described in our paper can be suitably modified if we use 4-connectedness for parallel automatic generation of such shape numbers.

e. Limitations of this approach: Finally a note of caution: in the method described here, we assume that the given object is digitized and the resulting polygonal boundary is examined for convexity. It may happen that new concavities are introduced in digitization (due to orientation and size of mesh) which may not exist in the original object. For example, the rotation of a square when digitized will exhibit concavities. This limitation, however, exists in other available methods also [16].

References

[1] Arcelli, C., and Cordella, L., Concavity point detection by iterative arrays, Computer Graphics Image Processing, 3, pp. 34-47, 1974.

[2] Hodes, L., Discrete approximation of continuous convex blobs, SIAM J. Appl. Math., 19, pp. 477-485, 1970.

[3] Klette, R., Parallel operations on binary images, Computer Science Center, University of Maryland, College Park, TR-822, November 1979.

[4] Pavlidis, T., Structural Pattern Recognition, Springer-Verlag, New York, 1976.

[5] Rosenfeld, A., Digital Topology, Amer. Math. Monthly 86, pp. 621-630, 1979.

[6] Rosenfeld, A., and Kak, A. C., Digital Picture Processing, Academic Press, New York, 1976.

[7] Rutovitz, D., An algorithm for in-line generation of a convex cover, Computer Graphics Image Processing, 4, pp. 74-78, 1975.

[8] Sankar, P. V., and Krishnamurthy, E. V., On the compactness of subsets of digital pictures, Computer Graphics Image Processing, 8, 136-143, 1978.

[9] Schachter, B., Decomposition of polygons into convex sets, IEEE Trans. Comp., C-27, pp. 1078-1082, 1978.

[10] Shoenberg, I. J., Isoperimetric inequality for closed curves convex in even dimensional Euclidean space, Acta Mathematica, 91, pp. 143-164, 1954.

[11] Sklansky, J., Chazin, R. L., and Hauser, B. J., Minimum perimeter polygons of digitized silhouettes, IEEE Trans. Comp., C-21, pp. 260-268, 1972.

[12] Sklansky, J., Measuring concavity on a rectangular mosaic, IEEE Trans. Comp., C-21, pp. 1355-1364, 1972.

[13] Sklansky, J., Recognition of convex blobs, Pattern Recognition, 2, pp. 3-10, 1970.

[14] Sklansky, J., On filling cellular concavities, Computer Graphics Image Processing, 4, pp. 236-247, 1975.

[15] Sklansky, J., Cordella, L. P., and Levialdi, S., Parallel detection of concavities in cellular blobs, IEEE Trans. Comp., C-25, pp. 187-196, 1976.

[16] Bribiesca, E. and Guzman, A., How to describe pure form and how to measure differences in shapes using shape numbers, Proc. IEEE Conf. on Pattern Recognition and Image Processing, Chicago, Illinois, pp. 427-436, August 1979.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFOSR-TR- 80-0138	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) ALGORITHMS FOR TESTING CONVEXITY OF DIGITAL POLYGONS		5. TYPE OF REPORT & PERIOD COVERED Interim
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Reinhard Klette, E. V. Krishnamurthy		8. CONTRACT OR GRANT NUMBER(s) AFOSR 77-3271
9. PERFORMING ORGANIZATION NAME AND ADDRESS Computer Vision Laboratory, Computer Science Center, University of Maryland, College Park, MD 20742		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 61102F 2304/A2
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Office of Scientific Research/NM, Bolling AFB, Washington, D. C. 20332		12. REPORT DATE December, 1979
		13. NUMBER OF PAGES 20
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Pattern recogintion
Image processing
Shape
Convexity
Polygons

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A simple approach based on Shoenberg's theorem is described to test whether a set of border points of a simply 4-connected digital picture is convex. The sequential implementation of this method is linear in the number of points; the parallel algorithm needs constant time only, using bitwise parallel Boolean operations and shifts on binary matrices. Suitable modifications of this approach can be used for decomposing two-dimensional objects into convex sets and for filling concavities.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)